References: -

[1] Krishnan G. P. “Unification and refactoring of clones,” in Proceedings of the IEEE Conference on Software Maintenance, Reengineering and Reverse Engineering (CSMR-WCRE), 2014

[2]. Roy C. K, “Comparison and evaluation of code clone detection techniques and tools: A qualitative approach,” Science of Computer Programming

[3]. Roy C “The vision of software clone management: Past, present, and future (keynote paper),” in Proceedings of the IEEE Conference on Software Maintenance, Reengineering and Reverse Engineering, 2014 Software Evolution 2014

[4] G¨ode N. “Studying clone evolution using incremental clone detection,” Journal of Software: Evolution and Process 2013

[5] Nguyen H. A. “Clone management for evolving software,” IEEE Transactions on Software Engineering, 2012

[6] Zibran M. “A constraint programming approach to conflict-aware optimal scheduling of prioritized code clone refactoring,” in Proceedings of the 11th IEEE International Working Conference on Source Code Analysis and Manipulation, 2011

[7] Mondal, C. “Automatic ranking of clones for refactoring through mining association rules,” in Proceedings of the IEEE Conference on Software Maintenance, Reengineering and Reverse Engineering, 2014 Software Evolution Week, 2014

[8] Higo Y, “A metric-based approach to identifying refactoring opportunities for merging code clones in a java software system,” Journal of Software Maintenance and Evolution: Research and Practice,2013

[9] Anvik, L, “Who should fix this bug?” in Proc. Int. Conf. Softw. Eng., 2006

[10] J Pearl, “Bayesian networks: A model of self-activated memory for evidential reasoning,” in Proc. 7th Conf. Cognitive Sci. Soc., 1988

[11]. Friedman D., “Bayesian network classifiers,” Mach. Learn., 1997

[12] Duala-Ekoko E. , “Tracking code clones in evolving software,” in Proc. Int. Conf. Softw. Eng.2007

[13] Krinke J “A study of consistent and inconsistent changes to code clones,” in Proc. 14th Working Conf. Reverse Eng, 2007

[14] Sazawal, D. Notkin “An empirical study of code clone genealogies,” in Proc. ACM SIGSOFT Symp. Found. Softw. Eng, 2005

[15] Baxter, A “Clone detection using abstract syntax trees,” in Proc. Int. Conf. Softw. Maintenance, 1998

[16] D. Hao, T. Lan, H. Zhang, C. Guo, and L. Zhang, “Is this a bug or an obsolete test?” in Proc. Eur. Conf. Object-Oriented Program, 2013

[17] S. Bellon, R. Koschke, G. Antoniol, J. Krinke, and E. Merlo, “Comparison and evaluation of clone detection tools,” IEEE Trans. Softw. Eng., vol. 33, no. 9, pp. 577–591, Sep. 2007

[18] H. A. Basit and S. Jarzabek. Detecting higher-level similarity patterns in programs. In ESEC/FSE, 2005

[19] Pearl, “Bayesian networks: A model of self-activated memory for evidential reasoning,” in Proc. 7th Conf. Cognitive Sci. Soc., 1988

[20] N. Friedman, D. Geiger, and M. Goldszmidt, “Bayesian network classifiers,” Mach. Learn, 1997

Websites

http://digitalcommons.unl.edu/libphilprac/

http://indianjournals.com/ijor.aspx?target=ijor:pjolis&type=home

http://ir.inflibnet.ac.in/

http://irjs.info/index.php/irjs/article/viewFile/11291/5726

http://jdhemumbai.gov.in/

http://jjtu.ac.in/

http://nopr.niscair.res.in/handle/123456789/66

http://shodhganga.inflibnet.ac.in/

http://trp.org.in/ajist_home.html

http://w